

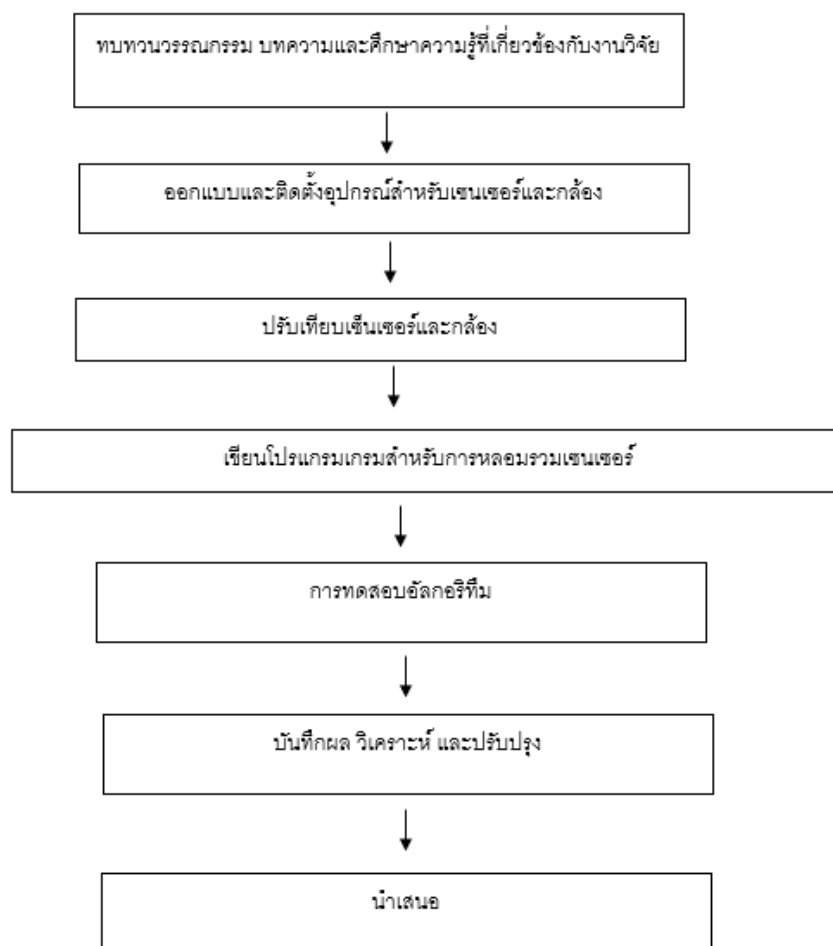
บทที่ 3

วิธีดำเนินการวิจัย

ในบทนี้จะนำเสนอรายละเอียดเชิงเทคนิคและวิธีการปฏิบัติทั้งหมด ซึ่งประกอบด้วย ขั้นตอนการดำเนินงานวิจัย การออกแบบและติดตั้งอุปกรณ์ การปรับเทียบเซนเซอร์และกล้อง การเขียนโปรแกรมสำหรับการหลอมรวมเซนเซอร์ และการทดสอบอัลกอริทึม โดยการดำเนินการเหล่านี้เป็นไปตามแผนผังภาพรวมที่แสดงไว้ เพื่อให้มั่นใจว่าระบบที่พัฒนาขึ้นจะสามารถทำงานได้อย่างแม่นยำและมีประสิทธิภาพภายใต้สภาวะการใช้งานจริงนอกอาคาร

3.1 ขั้นตอนการดำเนินงานวิจัย

งานวิจัยนี้เป็นการพัฒนาระบบระบุตำแหน่งของหุ่นยนต์อัตโนมัติที่ใช้สำหรับการเคลื่อนที่ระหว่างอาคาร โดยใช้ระบบกำหนดตำแหน่งบนโลกและกล้อง และทำการกรองข้อมูลจากเซนเซอร์ด้วยตัวกรองคาลมาล โดยมีวิธีการดำเนินการดังนี้



3.2 ออกแบบและติดตั้งอุปกรณ์สำหรับเซนเซอร์และกล้อง

การออกแบบอุปกรณ์สำหรับการติดตั้งเซนเซอร์และกล้องนั้นจะต้องมีการคำนึงถึงการจัดวางตำแหน่งท่าทางของตัวอุปกรณ์รวมไปถึงการต่อวงจรไฟฟ้าด้วย โดยกล้องและเซนเซอร์ที่เรานำมาใช้มีดังนี้

- 1) Intel RealSense D435 Camera เป็นกล้องที่รวมกล้อง RGB, กล้อง Depth เอาไว้ในอุปกรณ์เดียว มีความสามารถในการประมวลผลภาพและวัดระยะความลึก
- 2) MPU9250 Inertial Measurement Unit (IMU) เป็นอุปกรณ์ที่ใช้ตรวจจับการเคลื่อนไหวและทิศทาง โดยในตัว IMU นั้นมีเซนเซอร์ Accelerometer, Gyroscope และ Magnetometer
- 3) WTGPS 300P เป็นโมดูลนำทางที่รวมเทคโนโลยี GPS เข้าไว้ ช่วยให้สามารถทราบตำแหน่งของวัตถุบนพื้นโลก

แบบที่ได้จากการออกแบบด้วยโปรแกรม 3 มิติ จะถูกนำไปผลิตและขึ้นรูปชิ้นส่วนด้วยวัสดุที่เป็นพลาสติก PLA+ หลังจากประกอบชิ้นส่วนเรียบร้อยแล้วจะได้ใช้ในงานวิจัยดังรูปต่อไปนี้



รูปที่ 0.1 ประกอบที่หุ้มกล่อง

โดยรถที่นำมาทดสอบนั้นเป็นรถไฟฟ้าขนาดเล็ก 4 ล้อ มีขนาดโดยประมาณ ยาว 1.5 เมตร กว้าง 1 เมตร และสูง 2 เมตร มีลักษณะการขับเคลื่อนแบบ Ackerman steering โดยขับเคลื่อนด้วยล้อหลัง และใช้พวงมาลัยในการเลี้ยว ความเร็วสูงสุดอยู่ที่ 40 km/h โดยรถไฟฟ้าที่นำมาทดสอบลักษณะดังภาพ



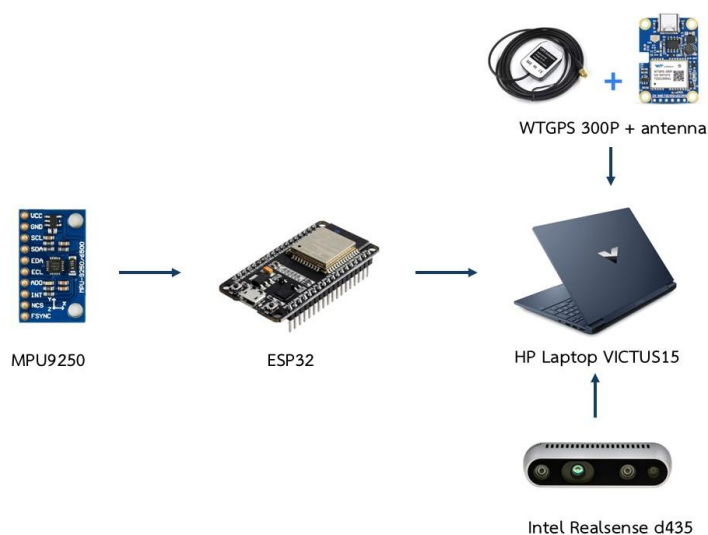
รูปที่ 0.2 ภาพรถไฟฟ้าขนาดเล็กที่นำมาใช้ทดสอบ

หลังจากเราได้ออกแบบและทำการประกอบชิ้นส่วนสำหรับการทดลองแล้ว เราจะนำเซนเซอร์และกล่องทั้งหมดไปติดตั้งบนรถโดยตำแหน่งการติดตั้งจากแสดงดังรูป



รูปที่ 0.3 ตำแหน่งการติดตั้งเซนเซอร์และกล้อง

โดยตำแหน่งในการติดตั้งนั้น กล้องจะถูกติดตั้งบริเวณหน้ารถเพื่อให้สามารถมองเห็นสภาพแวดล้อมด้านหน้า เซนเซอร์ความเฉื่อยติดตั้งไว้บริเวณตรงกลางของรถ และ GPS ถูกติดตั้งบริเวณด้านหลังของรถ เพื่อลดโอกาสในการเกิดสัญญาณรบกวนจากกล้อง โดยแผนผังการเชื่อมโยงส่วนประกอบต่าง ๆ จะแสดงดังในรูปที่ 3.3



รูปที่ 0.4 แผนผังการเชื่อมโยงส่วนประกอบต่าง ๆ ของอุปกรณ์

จากรูปที่ 3.4 ได้แสดงแผนภาพระบบเก็บข้อมูลและระบุตำแหน่ง โดยมีแหล่งข้อมูลสามชนิด คือ IMU, กล้อง RGB-D และ GPS แล้วส่งเข้าสู่คอมพิวเตอร์ศูนย์กลางเพื่อประมวลผลร่วมกัน ลูกศรแสดงทิศทางการไหลของข้อมูลจากเซนเซอร์ไปยังเครื่องประมวลผล ทางซ้ายคือโมดูล MPU9250 ทำหน้าที่วัดความเร่ง อัตราเชิงมุม และสนามแม่เหล็กแบบความถี่สูง เชื่อมต่อกับ ESP32 ผ่านบัส

I²C/SPI เพื่ออ่านค่าดิบและจัดรูปแบบเฟรมข้อมูล จากนั้น ESP32 ส่งข้อมูล IMU ไปยัง HP Victus 15 เพื่อใช้เป็นอินพุตของอัลกอริทึมประมาณท่าทางและตัวกรองกาลมาน ทางขวามือคือโมดูล WTGPS 300P พร้อมเสาอากาศ ซึ่งให้ตำแหน่งเชิงภูมิศาสตร์แบบสัมบูรณ์และเวลาอ้างอิง ส่งสตรีม NMEA/UBX เข้าสู่แล็ปท็อปทาง USB/UART เพื่อใช้เป็นฐานข้อมูลตำแหน่งระดับเมตรและแหล่งเวลาสำหรับซิงโครไนซ์ ส่วนล่างเป็นกล่อง Intel RealSense D435 ที่เชื่อมต่อ USB 3.0 กับแล็ปท็อป เพื่อให้ภาพ RGB และความลึกแบบเรียลไทม์ ใช้สำหรับวิชันโอโดเมทรี/SLAM และการสังเคราะห์แผนที่สภาพแวดล้อม HP Laptop Victus 15 ทำหน้าที่เป็นศูนย์รวมการประมวลผล รวมข้อมูลจาก IMU, GPS และ D435 เข้าด้วยกันเพื่อคำนวณตำแหน่งและทิศทาง

3.3 ปรับเทียบเซ็นเซอร์และกล้อง

การปรับเทียบเซ็นเซอร์และกล้องเป็นขั้นตอนสำคัญในระบบวิศวกรรมภาพ (Computer Vision) และหุ่นยนต์ (Robotics) เพื่อให้การรับรู้ภาพหรือข้อมูลจากเซ็นเซอร์มีความแม่นยำมากที่สุด โดยอุปกรณ์ที่เรานำมาปรับเทียบจะมีด้วยกันดังนี้

3.3.1 Intel RealSense D435 Camera

การปรับเทียบกล้องนั้นเป็นขั้นตอนในการแก้ความคลาดเคลื่อนและทำให้เราได้พารามิเตอร์ของกล้อง ที่จำเป็นต่อการคำนวณหาตำแหน่งจากฟิสิกส์ภาพโดยตรง หากไม่ทำการปรับเทียบข้อมูลจากกล้องจะเพี้ยนและส่งผลต่อ Visual Odometry โดยพารามิเตอร์ของกล้องที่สำคัญได้แก่ Intrinsic แต่ในกล้องของ Intel RealSense D435 นั้นถูกปรับเทียบมาจากโรงงานอยู่แล้วและไดรเวอร์ของกล้องจะเผยแพร่พารามิเตอร์ให้ครบเลย เพียงแค่เราติดตั้งไดรเวอร์และทำการเปิดกล้องสำหรับเช็คค่า โดยพารามิเตอร์ของกล้องนั้นจะเป็นดังสมการด้านล่าง

$$\text{intrinsics properties} = \begin{bmatrix} 639.31 & 0 & 637.47 \\ 0 & 639.31 & 362.69 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

3.3.2 Gyroscope

การปรับเทียบคือการคำนวณค่า offset สำหรับแต่ละแกนของไจโรสโคป โดยไจโรสโคปเป็นเซ็นเซอร์ที่สามารถปรับเทียบได้ง่ายที่สุด เนื่องจากค่าที่คาดว่าจะอ่านได้ภายใต้สภาวะนิ่งนั้นค่อนข้างชัดเจน โดยปกติแล้ว ทั้งสามแกนของไจโรสโคปจะแสดงค่าเป็น 0 องศาต่อวินาที เมื่อ IMU ไม่ได้เคลื่อนไหว วิธีการวัดค่า offset คือการอ่านค่าข้อมูลจากไจโรสโคปขณะที่ IMU อยู่ในสภาวะนิ่งแล้วนำค่าที่ได้มาใช้เป็น offset ในการหักลบจากค่าที่อ่านได้ในอนาคต เพื่อชดเชยความคลาดเคลื่อนโดยค่า offset ในแต่ละแกนเป็นดังนี้

$$Gx_{\text{offset}} = 0.01$$

$$Gy_{\text{offset}} = 0.02$$

$$Gz_{\text{offset}} = 0.01$$

3.3.3 Accelerometer

การปรับเทียบจำเป็นต้องใช้ประโยชน์จากแรงเร่งเนื่องจากแรงโน้มถ่วง ขณะที่เซ็นเซอร์อยู่นิ่งค่าที่วัดได้จะเท่ากับเวกเตอร์ของแรงโน้มถ่วงแต่กระจายบนแกน x, y และ z ขึ้นอยู่กับการวางทิศทาง เราเลยใช้แรงโน้มถ่วงเป็นค่าที่รู้ล่วงหน้าเพื่อปรับ offset ของแต่ละแกน โดยค่า offset ในแต่ละแกนเป็นดังนี้

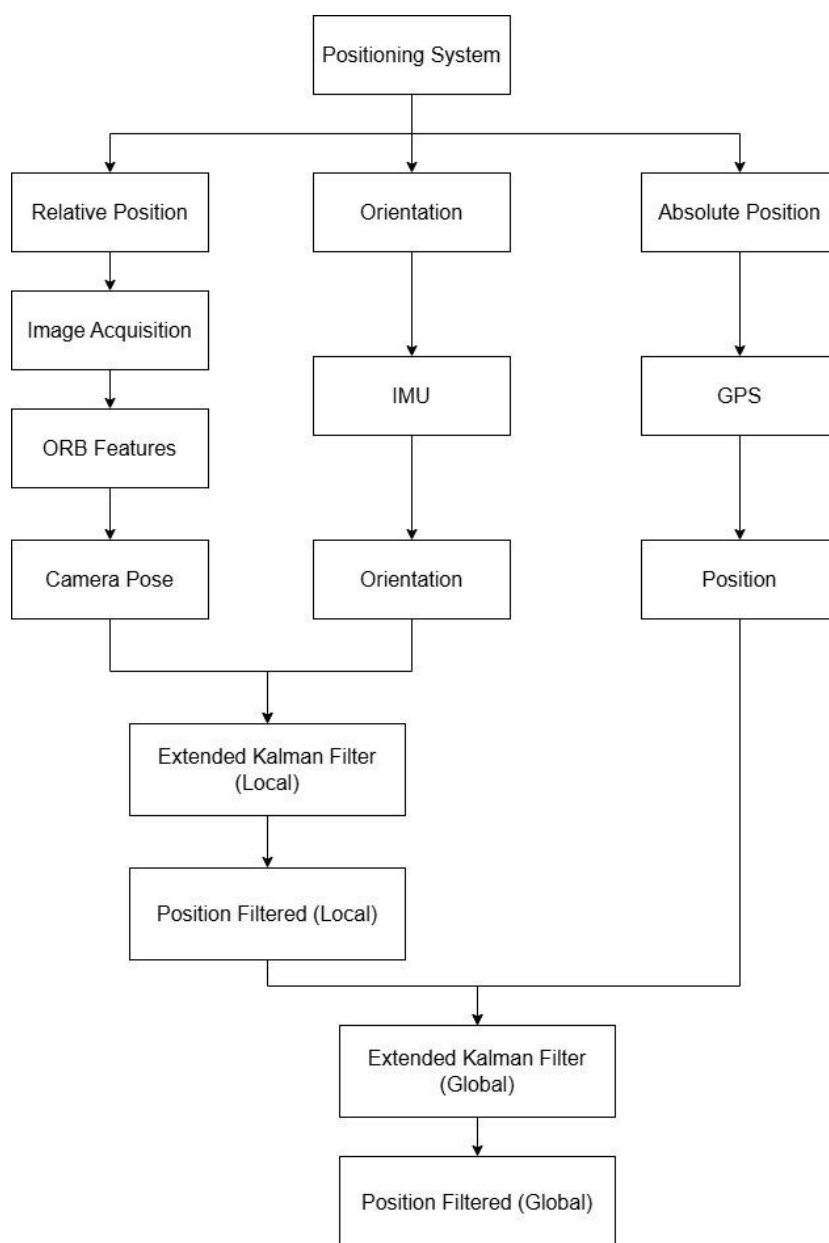
$$Ax_{\text{offset}} = 0.01$$

$$Ay_{\text{offset}} = 0.01$$

$$Az_{\text{offset}} = 9.81$$

3.4 เขียนโปรแกรมสำหรับการหลอมรวมเซนเซอร์

ขั้นตอนนี้จะเป็นการนำข้อมูลตำแหน่งและท่าทางที่ได้จาก ORB-SLAM ทำการหลอมรวมข้อมูลจาก IMU และ GPS โดยใช้ Extended Kalman Filter ในการประมาณสถานะใหม่ของตำแหน่ง ณ ปัจจุบัน ใช้ระบบปฏิบัติการหุ่นยนต์ (ROS) สำหรับเป็นเครื่องมือในการหลอมรวม โดยใช้ robot localization package มีผังการทำงานโดยรวมของระบบดังรูปที่ 3.5

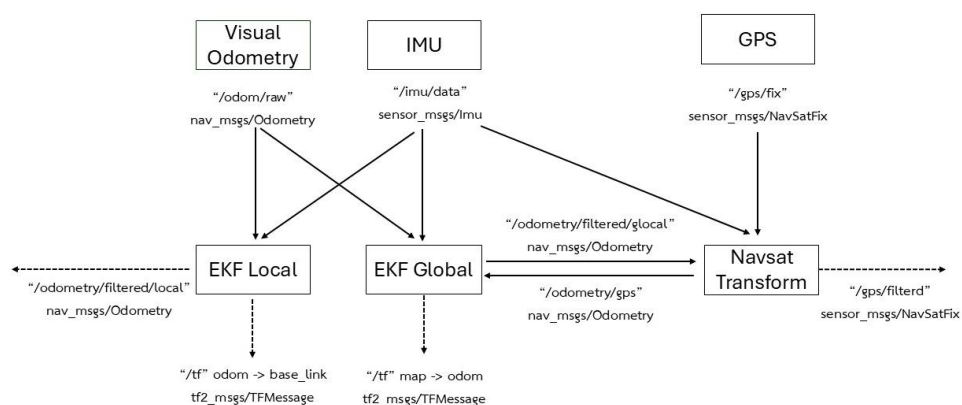


รูปที่ 0.5 แผนผังภาพรวมของระบบหลอมรวมเซนเซอร์

แผนผังการทำงานแสดงให้เห็นถึงลำดับขั้นตอนทั้งหมดของระบบหลอมหลวมเซนเซอร์ สำหรับระบบระบุตำแหน่งที่ใช้เซนเซอร์หลายชนิดร่วมกัน ได้แก่ กล้อง, IMU และ GPS โดยมี จุดมุ่งหมายหลักคือการประมาณค่าตำแหน่งและทิศทางให้มีความแม่นยำสูง ราบรื่นต่อเนื่อง ระบบหลักในการระบุตำแหน่ง รับข้อมูลจากแหล่งข้อมูลหลักสามส่วน ได้แก่ ตำแหน่งสัมพัทธ์ (Relative Position), ทิศทางการเคลื่อนที่ (Orientation) และตำแหน่งสัมบูรณ์ (Absolute Position) ในส่วนของ Relative Position จะเริ่มจากกระบวนการ Image Acquisition ซึ่งเป็นการรับข้อมูลภาพจาก

กล้องแบบเรียลไทม์ ข้อมูลภาพเหล่านี้จะถูกนำไปประมวลผลเพื่อดึงเอาคุณลักษณะ (Features) ที่สำคัญออกมา โดยในระบบนี้ใช้เทคนิค ORB Features (Oriented FAST and Rotated BRIEF) จากนั้นข้อมูลคุณลักษณะจะถูกนำไปคำนวณเพื่อหาท่าทางของกล้อง (Camera Pose) ซึ่งเป็นตำแหน่งและการหมุนของกล้องในกรอบอ้างอิงสัมพัทธ์ ในขณะเดียวกัน ส่วนของ Orientation จะเริ่มจากข้อมูลที่ได้จาก IMU (Inertial Measurement Unit) ซึ่งเป็นเซนเซอร์ที่สามารถวัดความเร่งเชิงเส้นและอัตราการหมุนของหุ่นยนต์ได้โดยตรง ข้อมูลเหล่านี้ถูกประมวลผลเพื่อให้ได้ทิศทาง (Orientation) ของหุ่นยนต์ ในส่วนของ Absolute Position ข้อมูลจะมาจาก GPS (Global Positioning System) ซึ่งให้ตำแหน่งในรูปแบบพิกัดภูมิศาสตร์ ข้อมูลนี้มีข้อดีคือไม่เกิดการดริฟต์ แต่มีข้อเสียคือมีความถี่ต่ำและอาจมีนอยส์สูงจากสภาพแวดล้อม เมื่อข้อมูลจากทั้งสามส่วนถูกคำนวณเรียบร้อยแล้ว ระบบจะเข้าสู่กระบวนการรวมข้อมูลในขั้นตอนของ Extended Kalman Filter (Local) ซึ่งจะรวมข้อมูลจาก Camera Pose (ตำแหน่งสัมพัทธ์) เข้ากับข้อมูล Orientation จาก IMU เพื่อสร้างตำแหน่งที่ผ่านการกรองแล้วในกรอบอ้างอิงท้องถิ่น (Position Filtered Local) ขั้นตอนนี้มีเป้าหมายเพื่อให้ได้ผลลัพธ์ที่ราบรื่นและต่อเนื่อง ข้อมูลที่ผ่านการกรองในขั้นตอน Local แล้ว จะถูกส่งเข้าสู่ขั้นตอนถัดไปคือ Extended Kalman Filter (Global) ซึ่งจะทำหน้าที่ผสานข้อมูลจาก Position Filtered (Local) เข้ากับข้อมูลตำแหน่งจาก GPS ขั้นตอนนี้ทำให้ระบบสามารถแก้ไขดริฟต์ที่อาจเกิดขึ้นจากขั้นตอน Local ได้ และทำให้ตำแหน่งสุดท้ายที่ได้ (Position Filtered Global) มีทั้งความต่อเนื่อง ความราบรื่น

การที่จะสามารถหลอมหลวมข้อมูลเซนเซอร์ได้ดังแผนภาพนั้น เพื่อง่ายสำหรับการทดสอบ และเขียนโปรแกรม ผู้วิจัยได้ใช้เครื่องมือที่ชื่อว่าระบบปฏิบัติการหุ่นยนต์เป็นเครื่องมือหลักในการทำวิจัยครั้งนี้ โดยจะมีแผนผังการทำงานของ node และ topic ต่างๆ รวมถึงการทรานส์ฟอร์มข้อมูลจากเฟรมต่างๆ ดังแผนภาพในรูปที่ 3.6

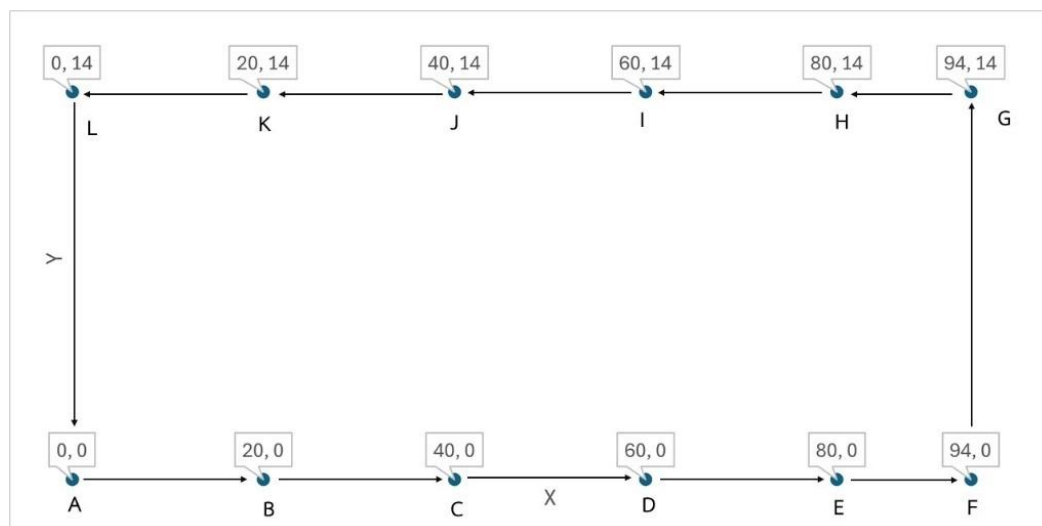


รูปที่ 0.6 แผนภาพโครงสร้างการรวมข้อมูลเซนเซอร์ใน ROS เพื่อระบุตำแหน่ง

ภาพที่แสดงเป็นสถาปัตยกรรมของระบบการประมวลผลข้อมูลตำแหน่งและท่าทางของหุ่นยนต์เคลื่อนที่ใน ROS ซึ่งรวมเอาข้อมูลจากหลายแหล่งเซนเซอร์เข้าด้วยกันผ่านกระบวนการ sensor fusion เพื่อให้ได้ตำแหน่งที่แม่นยำและต่อเนื่อง ภาพเริ่มจากสามแหล่งข้อมูลหลักคือ Visual Odometry, IMU และ GPS ซึ่งส่งข้อมูลออกมาในรูปแบบข้อความของ ROS แต่ละชนิด โดย Visual Odometry ส่งข้อมูลโอโดเมทรีสัมพัทธ์ผ่าน topic “/odom/raw” ในรูปแบบ nav_msgs/Odometry ส่วน IMU ส่งข้อมูลการเร่งและอัตราการหมุนผ่าน topic “/imu/data” ในรูปแบบ sensor_msgs/Imu และ GPS ส่งพิกัดสมบูรณ์ผ่าน topic “/gps/fix” ในรูปแบบ sensor_msgs/NavSatFix ข้อมูลจาก Visual Odometry และ IMU จะถูกป้อนเข้าสู่ EKF Local ซึ่งเป็นฟิลเตอร์คาลมานแบบขยายที่รวมข้อมูลเซนเซอร์เพื่อให้ได้โอโดเมทรีที่เรียบและต่อเนื่องในกรอบอ้างอิงท้องถิ่น โดยจะเผยแพร่ผลลัพธ์ออกมาเป็น topic “/odometry/filtered/local” และเผยแพร่ทรานส์ฟอร์มระหว่างกรอบ odom และ base_link ผ่าน tf2_msgs/TFMessage ต่อมาข้อมูลทั้งจาก Visual Odometry และ IMU จะถูกส่งต่อไปยัง EKF Global เช่นกัน โดยมีหน้าที่รวมข้อมูลในระดับกรอบอ้างอิงโลกและสร้างผลลัพธ์เป็น topic “/odometry/filtered/global” ซึ่งเผยแพร่การทรานส์ฟอร์มระหว่างกรอบ map และ odom เอาต์พุตจาก EKF Global จะถูกนำไปใช้ร่วมกับข้อมูลจาก GPS ภายในโหนด Navsat Transform ซึ่งทำหน้าที่แปลงข้อมูลพิกัดจากระบบภูมิศาสตร์ให้เชื่อมโยงกับกรอบอ้างอิงของหุ่นยนต์อย่างถูกต้อง ในขั้นตอนนี้ข้อมูล /odometry/filtered/global จะถูกรวมกับ “/gps/fix” เพื่อสร้าง topic “/odometry/gps” ในรูปแบบ nav_msgs/Odometry และข้อมูล “/gps/filtered” ในรูปแบบ sensor_msgs/NavSatFix ซึ่งเป็นข้อมูลตำแหน่งที่ผ่านกระบวนการ fusion แล้วและพร้อมใช้งานในระบบนำทางหรือการควบคุม โครงสร้างการเชื่อมโยงข้อมูลทั้งหมดนี้ทำให้หุ่นยนต์สามารถใช้ประโยชน์จากข้อดีของแต่ละเซนเซอร์ได้อย่างเต็มที่ โดย Visual Odometry ให้ความละเอียดสูงในระยะใกล้ IMU ให้การตอบสนองที่รวดเร็ว และ GPS ให้ตำแหน่งสมบูรณ์ที่อ้างอิงกับโลกจริง กระบวนการผ่าน EKF และ Navsat Transform จึงเป็นขั้นตอนสำคัญที่ทำให้ข้อมูลทั้งหมดถูกรวมกันอย่างเหมาะสม เพื่อให้หุ่นยนต์มีความเข้าใจตำแหน่งที่ต่อเนื่องและถูกต้องตลอดเวลาในระบบ ROS

3.4 การทดสอบอัลกอริทึม

หลักจากที่เราทำการเขียนโปรแกรมสำหรับการหลอมรวมเซนเซอร์ และทำการทดสอบและประเมินค่าดูแล้ว จากนั้นอุปกรณ์จะถูกนำไปติดตั้งและทดสอบกับสภาพแวดล้อมภายนอกอาคาร บริเวณ ที่จอดรถอาคารเครื่องมือ F16 มหาวิทยาลัยเทคโนโลยีสุรนารี เส้นทางประกอบด้วยดังต่อไปนี้ ดังรูปที่ 3.7



รูปที่ 0.7 เส้นทางสำหรับการทดสอบ

ภาพนี้เป็นแผนที่เส้นทางทดสอบแบบลูบสี่เหลี่ยมผืนผ้าในพิกัดคาร์ทีเซียน โดยกำหนดให้จุดกำเนิดอยู่ที่มุมซ้ายล่าง A ซึ่งมีพิกัด (0, 0) แกน X พาดไปทางขวาและแกน Y พาดขึ้นด้านบนบน โหนดตามขอบล่างเรียงจากซ้ายไปขวาเป็น $A(0, 0) \rightarrow B(20, 0) \rightarrow C(40, 0) \rightarrow D(60, 0) \rightarrow E(80, 0) \rightarrow F(94, 0)$ จากนั้นขึ้นแนวตั้งขวาสู่ $G(94, 14)$ แล้ววิ่งตามขอบบนย้อนกลับทางซ้ายผ่าน $H(80, 14) \rightarrow I(60, 14) \rightarrow J(40, 14) \rightarrow K(20, 14) \rightarrow L(0, 14)$ และปิดลูบโดยลงแนวตั้งซ้ายกลับมายัง $A(0, 0)$ ระยะช่วงแนวนอนหลักแบ่งเป็นช่วงละประมาณ 20 เมตร ยกเว้นช่วงปลายที่ 14 เมตรให้ครบความยาว 94 เมตร ส่วนแนวตั้งทั้งสองด้านสูง 14 เมตร ลูกศรบนเส้นแสดงทิศการเคลื่อนที่ในแต่ละด้านคือ ด้านล่างเคลื่อนที่ไปทางขวา ด้านขวาเคลื่อนที่ขึ้น-ลงตามที่กำหนดเพื่อเชื่อมกับด้านบนและด้านบนเคลื่อนที่ไปทางซ้ายก่อนลงกลับจุดเริ่มต้น ลูบนี้จึงเป็นเส้นทางมาตรฐานสำหรับเก็บข้อมูลและประเมินอัลกอริทึมระบุตำแหน่ง โดยมีจุดอ้างอิงพิกัดคงที่ที่ทุกมุมและทุกระยะ 20 เมตรเพื่อใช้เทียบความคลาดเคลื่อนของวิธีต่าง ๆ ได้อย่างชัดเจน โดยมีการทดสอบที่ความเร็วประมาณ 2 m/s และ 1 m/s ตามลำดับ

โดยสรุปแล้วบทนี้ได้นำเสนอวิธีดำเนินการวิจัยทั้งหมด ตั้งแต่การออกแบบโครงสร้างเชิงกล, การคัดเลือกและติดตั้งเซนเซอร์หลัก การปรับเทียบเซนเซอร์เพื่อลดความคลาดเคลื่อน, ไปจนถึงการออกแบบสถาปัตยกรรมซอฟต์แวร์สำหรับการหลอมรวมข้อมูลหัวใจสำคัญของวิธีการนี้ คือการใช้ Extended Kalman Filter สองระดับ (Local และ Global) ภายในระบบปฏิบัติการหุ่นยนต์ (ROS) เพื่อผสานจุดเด่นของแต่ละเซนเซอร์ ความละเอียดสูงของ Visual Odometry/ORB-SLAM การตอบสนองที่รวดเร็วของ IMU , และ ตำแหน่งสมบูรณ์ที่แก้ไขได้ของ GPS วิธีการนี้ถูกกำหนดให้ทดสอบบนรถไฟฟ้านาฬเล็กตามเส้นทางสี่เหลี่ยมผืนผ้าแบบลูบที่กำหนดไว้ชัดเจน ด้วยความเร็ว

ที่แตกต่างกัน ในบทถัดไปจะนำเสนอผลการทดสอบอัลกอริทึมตามขั้นตอนที่ระบุในบทนี้ โดยจะมีการวิเคราะห์และประเมินประสิทธิภาพของระบบระบุตำแหน่งที่พัฒนาขึ้น ในแง่ของความแม่นยำและความต่อเนื่องของตำแหน่งเปรียบเทียบกับพิกัดอ้างอิงของเส้นทางทดสอบ